Securing Data Across the Enterprise using SOA

Section II
EKMI Architecture

OASIS Open Standards
April 28, 2008

Arshad Noor, StrongAuth, Inc.
arshad.noor@strongauth.com

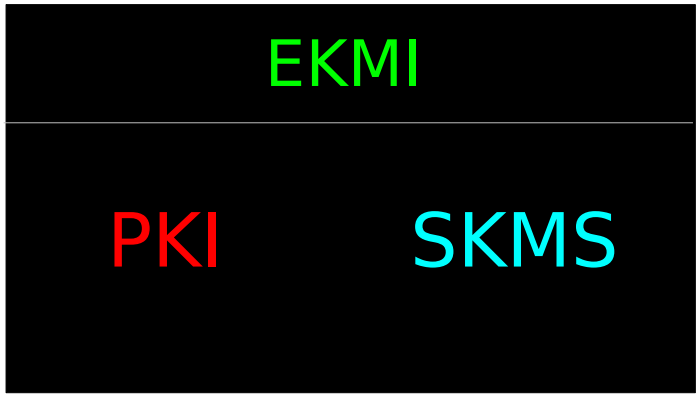
EKMI and its Components

What is an EKMI?

- An **Enterprise Key Management Infrastructure** is:

  "A collection of technology, policies and procedures for managing the life-cycle of ***all*** cryptographic keys in the enterprise."

EKMI Components



- Public Key Infrastructure (PKI)
- Symmetric Key Management System (SKMS)

What is an EKMI?

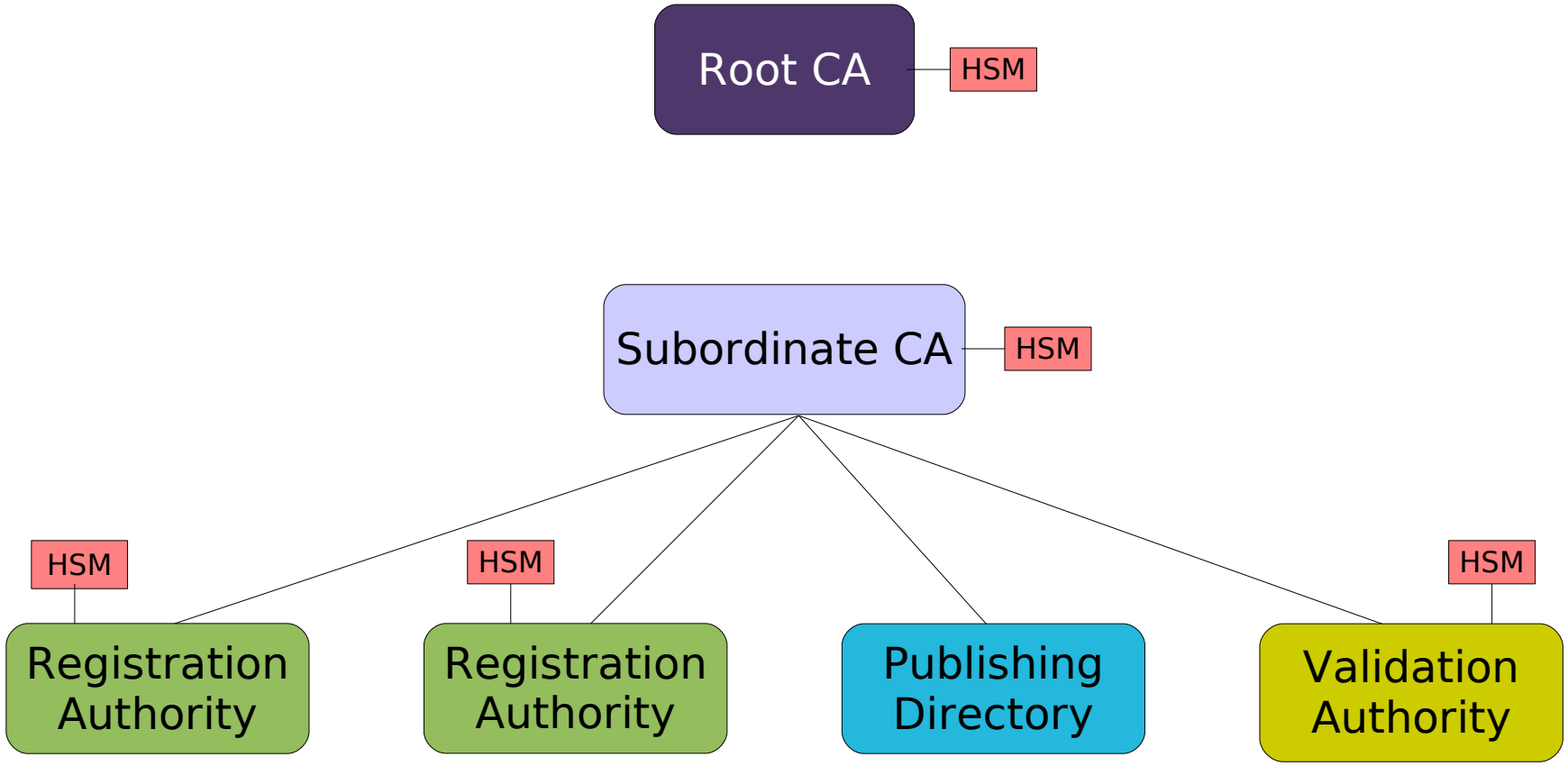
- PKI

  "A collection of technology, policies and procedures for managing the life-cycle of ***asymmetric*** cryptographic keys in the enterprise."

- SKMS

  "A collection of technology, policies and procedures for managing the life-cycle of ***symmetric*** cryptographic keys in the enterprise."

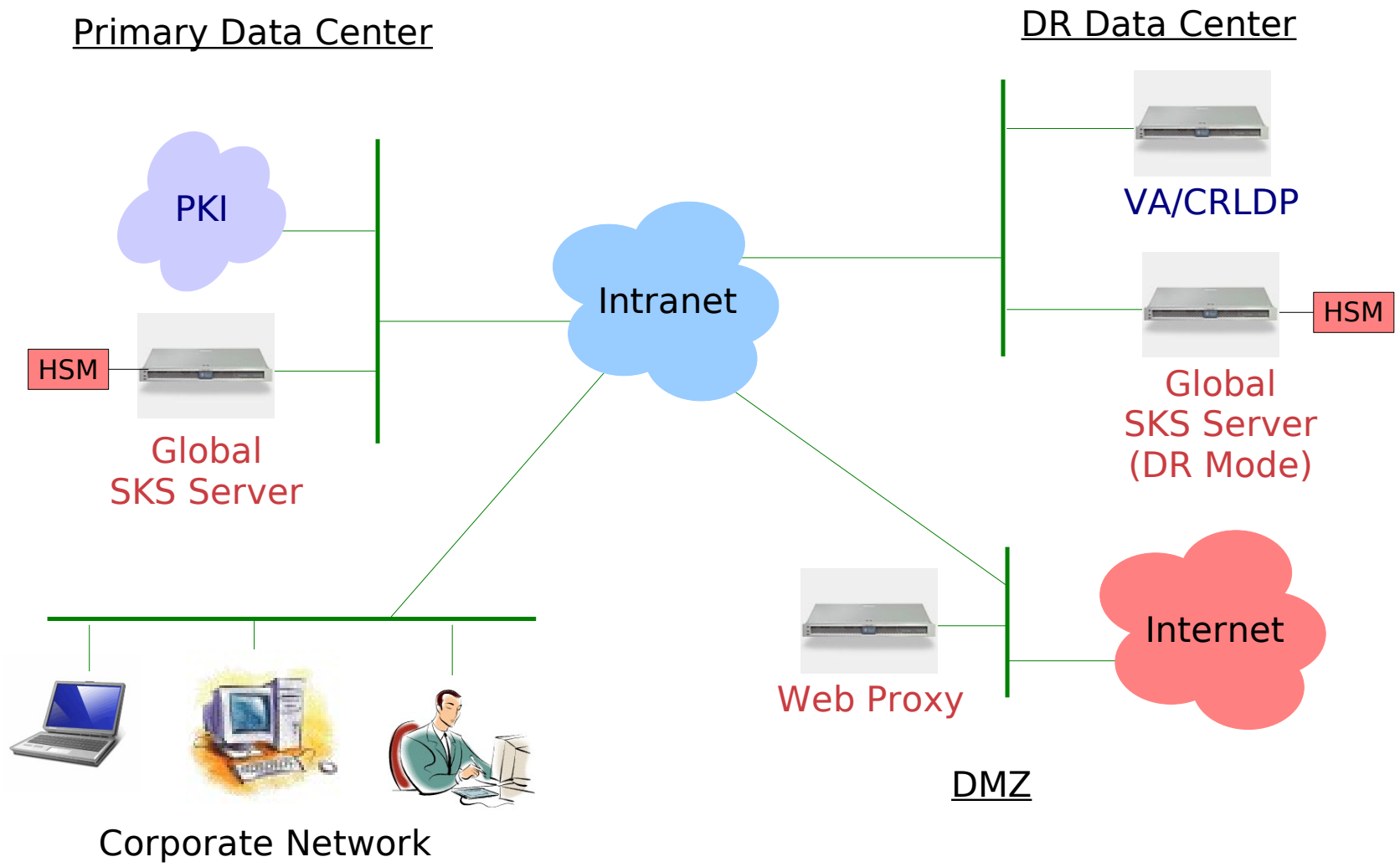
Typical PKI



Critical Components

- Cryptographic Token
  - Module where cryptographic keys are stored
- Hardware Security Module (HSM)
  - A cryptographic token that is implemented in hardware, and which performs cryptographic processing inside the hardware
- (US) Federal Information Procurement Standards (FIPS) 140-2
  - A conformance standard for cryptographic tokens and HSM's

Typical SKMS



Global SKS Server

- One per enterprise
- Define all SKMS objects here:
  - Clients, Servers, Client Groups, Key Groups, Key Use Policies, Key Cache Policies, Grants
- DR Mode GSKS server is identical but Read-Only*
- CPU-intensive; quad-core recommended
- HSM critical to security of server

SKS Server

- Any number per enterprise, as needed
  - One per continent recommended for global enterprises
- Configured to replicate to GSKS*
- CPU-intensive; quad-core recommended
- HSM critical to security of server

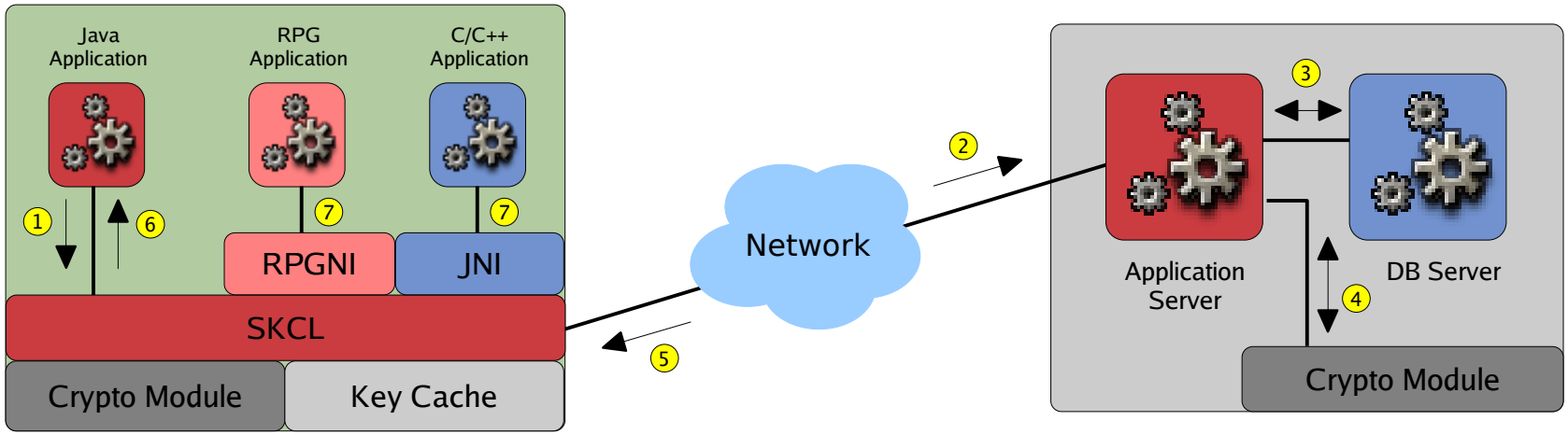
SKMS Client

- Unlimited number
- Maintains a list of SKS servers to get KM services from:
  1) Nearest SKS server on network
  2) GSKS Server
  3) GSKS DR-Mode Server
- Smartcard token or TPM chip strongly recommended for security

SKMS Client Examples

- Database servers
- Web Application servers
- Network File servers
- Desktops/Laptops
- Automated Teller Machines (ATM)
- Point-of-Sale (POS) Registers
- Personal Digital Assistant (PDA)
- Smart mobile devices: Banking, Healthcare

The Big Picture



1. Client Application makes a request for a symmetric key
2. SKCL makes a digitally signed request to the SKS
3. SKS verifies SKCL request, generates, encrypts, digitally signs & escrows key in DB
4. Crypto HSM provides security for RSA Signing & Encryption keys of SKS
5. SKS responds to SKCL with signed and encrypted symmetric key
6. SKCL verifies response, decrypts key and hands it to the Client Application
7. Native (non-Java) applications make requests through Java Native Interface

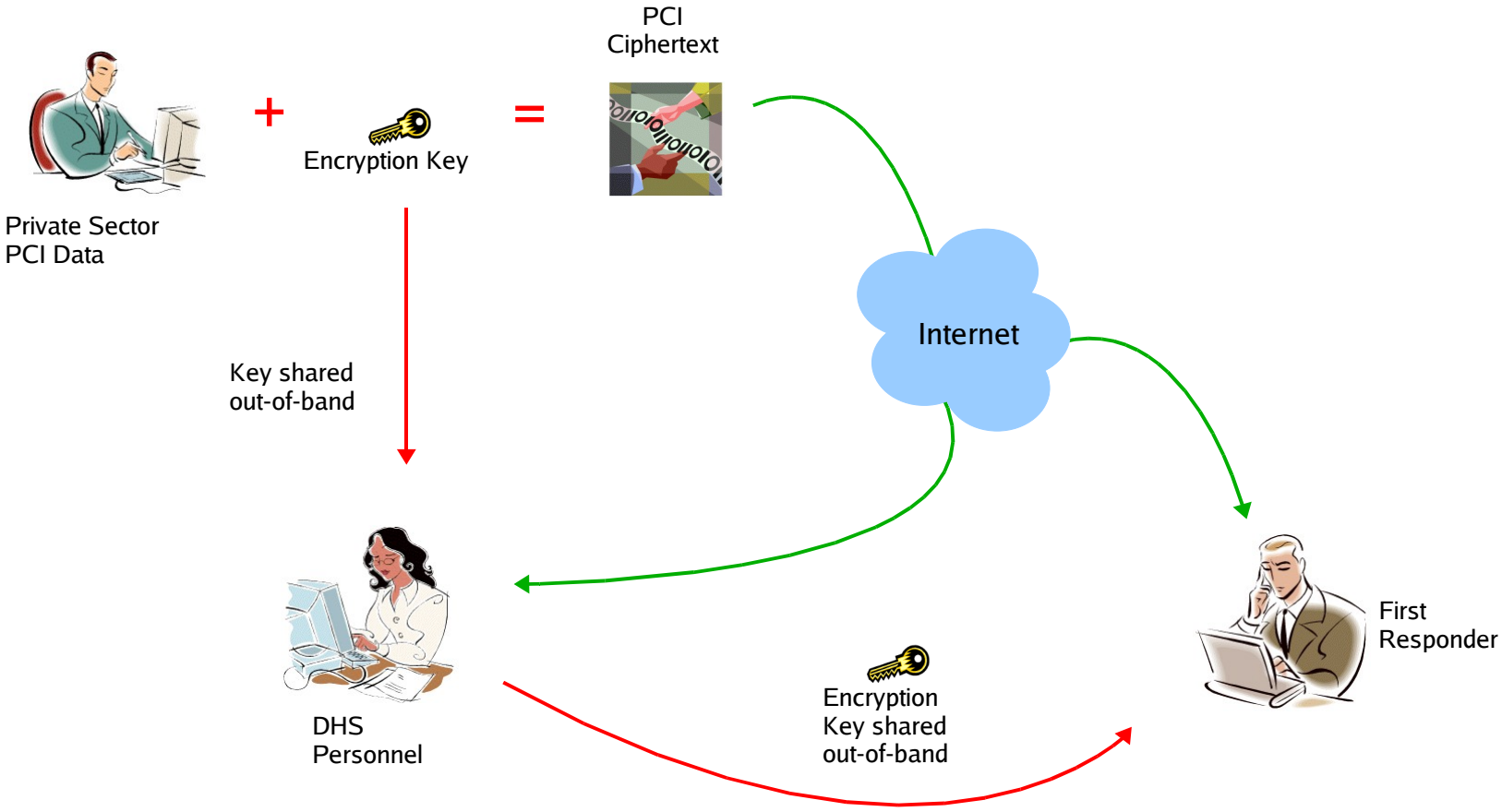
StrongKey® Securing the Core

SKMS Security

- Every request/response is digitally signed
- Every response is encrypted
- Every object in database is digitally signed
- All symmetric keys in cache are digitally signed and encrypted
- All crypto code is abstracted
  - FIPS 140-2 devices are easily integrated
- Administration console does not use UserID and Passwords; only SSL Client Auth.

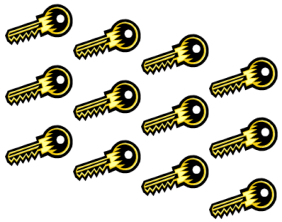
SKMS Use-Cases

Secure Data Sharing



Key-sharing Problem

Private Sector PCI Data
(Tens of thousands?)

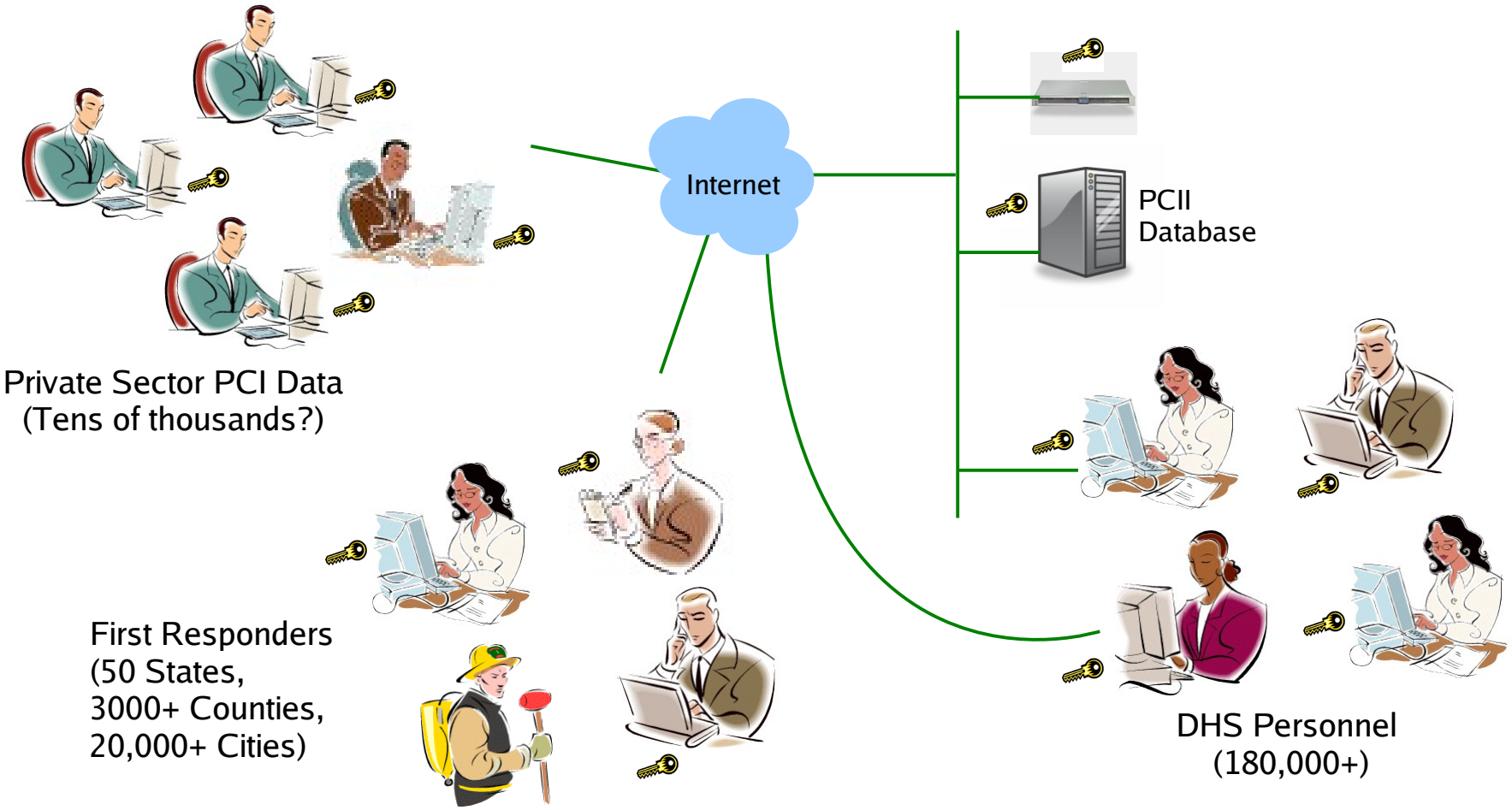


Encryption Keys

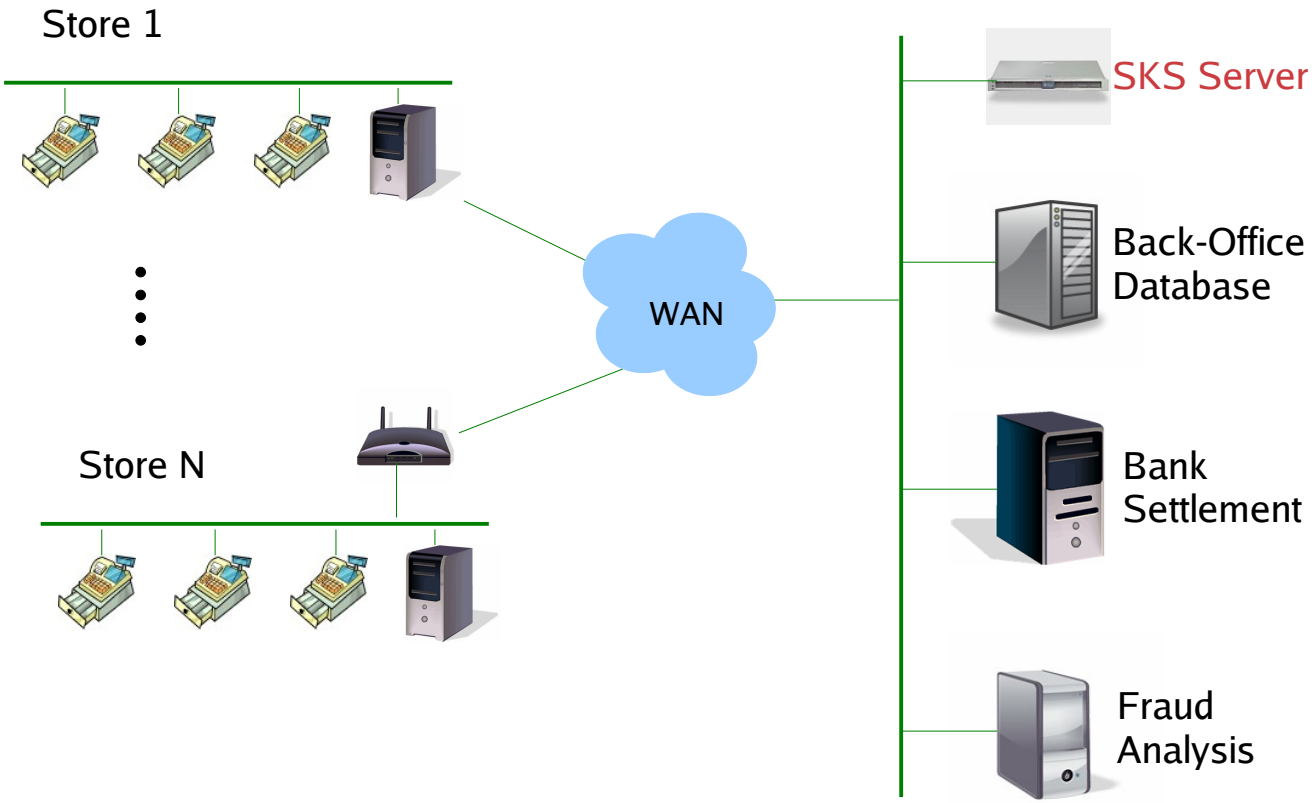
DHS Personnel
(180,000+)

First Responders
(50 States,
3000+ Counties,
20,000+ Cities)

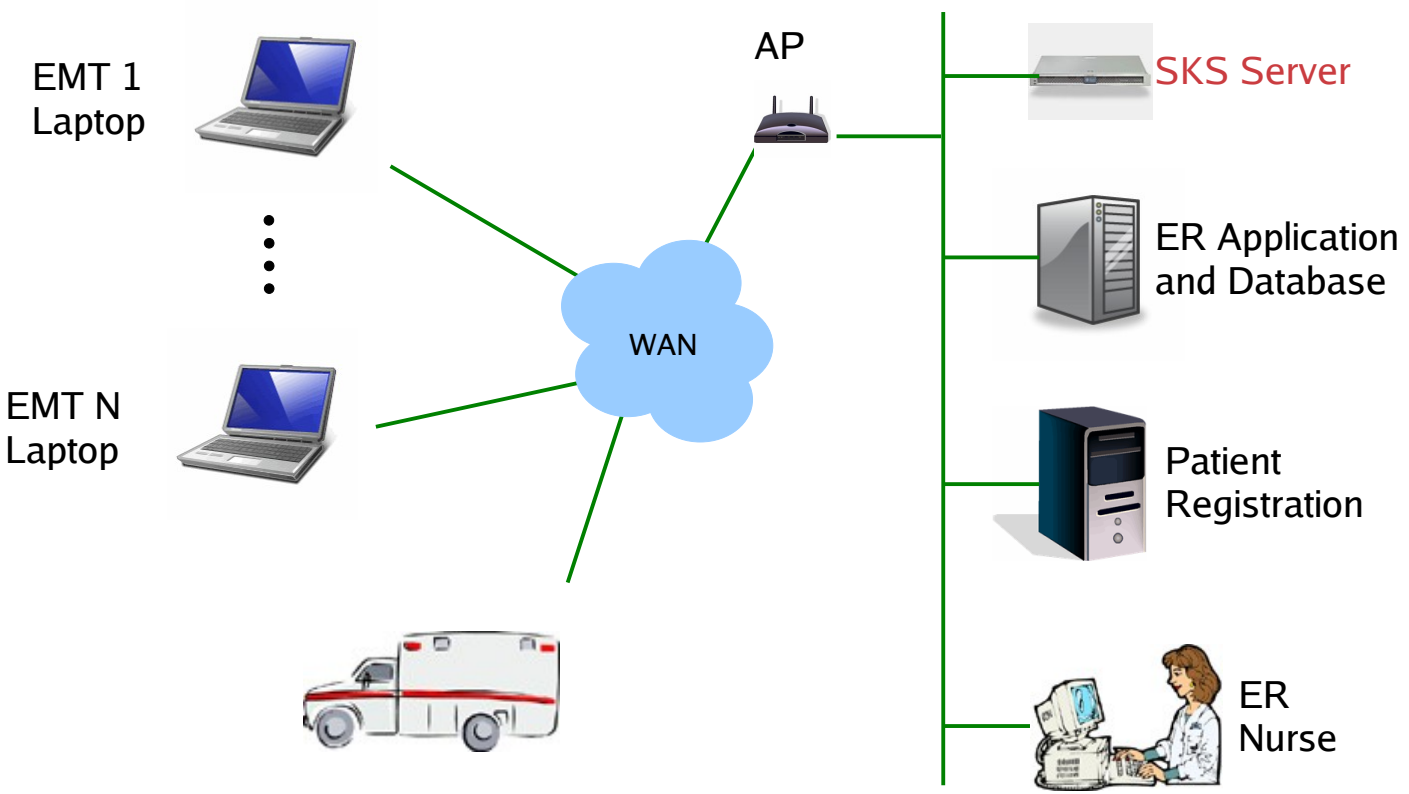
Solution



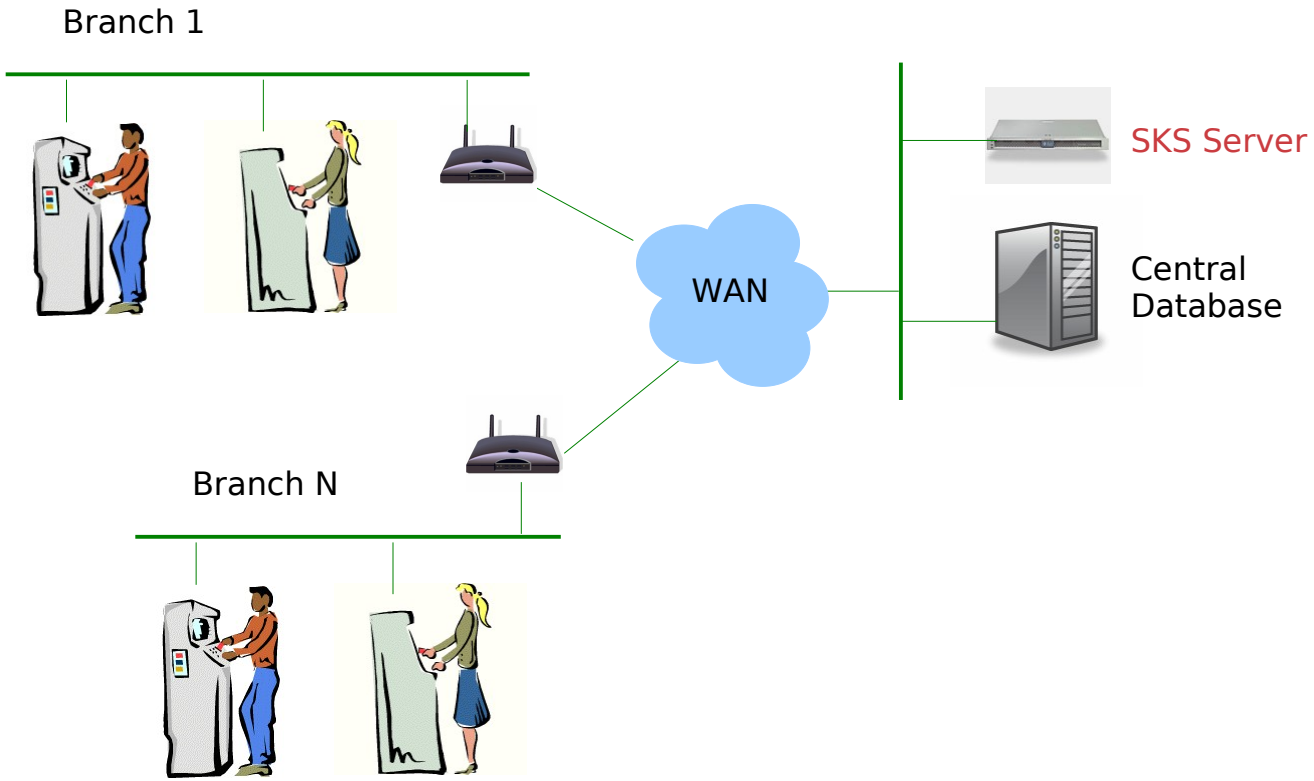
Retail



Healthcare



Finance



Conclusion

- Questions?
- Contact Information
  - www.strongauth.com
  - www.strongkey.org
  - info@strongauth.com
  - (408) 331-2000